[POSTER] Deformed Reality: Proof of concept and preliminary results

Nazim Haouchine, Antoine Petit, Frederick Roy, Stephane Cotin*

Inria, Mimesis Group, Strasbourg, France

Abstract

We introduce "Deformed Reality", a new paradigm to interactively manipulate objects in a scene in a deformable manner. Using the core principle of augmented reality to estimate rigid pose over time, our method enables the user to deform the targeted object while it is being rendered with its natural texture, giving the sense of a real-time object editing in user environment. The presented results show that our method can open new ways of using augmented reality by not only augmenting the scene but also interacting with it in a non-rigid manner.

Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems - Artificial, augmented, and virtual realities—;

1 Introduction

The considerable technical advances in Augmented Reality (AR) made it possible for users to have different sorts of interactions such as object grasping, rigid transform or surface deformation [1]. However, most of the time, a 3D mesh is superimposed using computer-generated texturing, making the final output unrealistic. In the context of image editing, recent methods using 3D stock models to replace the actual object projection in the image have been proposed [2]. These methods give the possibility of editing objects in a scene in 3D and extend the range of manipulation. However, they consider the camera to be fixed making it difficult to be used for AR purpose where the pose has to be estimated.

We aim at enabling users to deform objects in their environment without producing visual breaks due to 3D overlays. To this end, we introduce the concept of "Deformed Reality", a method for deforming objects in scene interactively. This method relies on a simultaneous and sequential rigid pose estimation, thanks to the use of object silhouette and image features and a real-time deformation computation, using physics-based modeling, that permits consistent non-rigid manipulations. The final composition is rendered using the original object texture making the mesh superimposition realistic. To the best of our knowledge, no similar method has yet been proposed.

2 Method

As shown in the pipeline of Figure 1, given an image, a 3D model and an initial pose, our method is separated onto two simultaneous processes: (1) estimating object pose by tracking in a rigid manner the object over frames and (2) computing deformation from user manipulation using an underlying physical model. Sequentially, the deformed 3D model is projected onto an inpainted frame following the estimated pose and re-textured using the original image-texture.

2.1 Model-based Rigid Pose Estimation

The issue of estimating the complete rigid 3D pose of the camera, with respect to the considered object along the image stream, is addressed based on the 3D model-based approach presented in [4]. The method consists on combining, in one optimization scheme, a geometrical information provided by the distances between model and image edges, with a dense color information through object/background color separation statistics along the model edges, and to take advantage of GPU acceleration to efficiently handle 3D models of any shape. In a nutshell, given a 3D

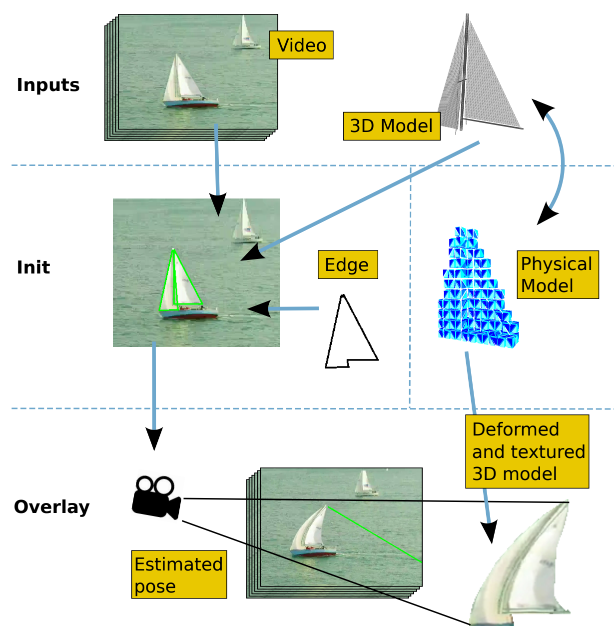

Figure 1: Deformed Reality pipeline: using as inputs a video and a 3D model corresponding to an object in the scene, the user first initially aligns the 3D model on the image, the first pose is estimated along with the object edges. They will be used to estimate in a rigid manner object pose over frames. Simultaneously, a physical model is built form the 3D model to enable deformable user interactions. The deformed mesh is textured and projected on an inpainted image using the estimated pose to produce the final composition.

model of the object, the goal is to estimate the camera pose $\mathbf{P}$ by minimizing, with respect to $\mathbf{P}$, the function Δ accounting for errors $e_i(\mathbf{P})$ between a set of visual features extracted from the image and the forward projection of their 3D homologous in the image plane:

$$\Delta(\mathbf{P}) = \sum_i \rho(e_i(\mathbf{P})) \tag{1}$$

where ρ is a robust estimator, which reduces the sensitivity to outliers. This is a non-linear minimization problem with respect to the pose parameters $\mathbf{P}$, and we follow a Gauss-Newton minimization framework to handle it. In order to benefit from the complementarity of different visual features and to overcome the limitations of classical single cue approaches, we integrate in the computation of Δ a geometrical edge-based error function Δ^g and a color-based one Δ^c with the corresponding weights w^g and w^c:

$$\Delta = w^g \Delta^g + w^c \Delta^c \tag{2}$$

*e-mail:stephane.cotin@inria.fr

Δ^g and Δ^c are computed in a similar way to [5]. Δ^g relies on line-to-point correspondences between model and image edges and on geometrical distances accounting for these correspondences. Edges have the advantage of being robust to illumination conditions but suffer from having similar appearance, resulting in ambiguities and potential local minima. For Δ^c, the idea is to avoid any image extraction or segmentation that could lead to outliers and mismatch. By modeling the color (or luminance) appearance on both sides of the edges of the projected 3D model using simple statistics, a better accuracy can be achieved, with the advantage of being robust to image or motion blur, background clutter or noise.

2.2 Deformable Manipulation

We rely on physics-based modeling to enable deformable manipulation [3]. Using the input 3D model M, we first build (at initialization) a volume $\mathbf{V}$ composed of co-rotated thetrahedral elements. These elements represent the Degrees-of-Freedom (DoF) of the physical model following a finite element representation.

Since a particular object deformation is specified by the displacements of nodal positions and the nodal forces. We build a stiffness matrix $\mathbf{K}$ depending on object's material properties, Young's modulus and Poisson's ratio. This enables us to linearize the relationship between nodal forces and nodal positions and permits real-time computation.

Finally, fixed boundary conditions $\mathbf{q}$ are necessary to correctly model the deformations. These constraints represent a part of the 3D model that is considered fixed.

We denote $\Theta(\cdot)$ as the function that permits to build a physical model D the input 3D model M so that:

$$\mathsf{D} = \Theta(\mathsf{M}, \mathbf{V}, \mathbf{K}, \mathbf{q}) \tag{3}$$

Once the volume built following $\Theta(\cdot)$, the stiffness defined and the constrained regions chosen, the user can interact with the object at each frame of the sequence by moving the DoF producing forces that will deform the object according to its stiffness.

2.3 Composition

Producing the final composition implies to continuously build a new image J as a result of the projection of the deformed and textured 3D model on the input image I. Denoting $\mathbf{T}$ the texture of the targeted object, the output composite image is built following the expression

$$\mathsf{J}^f = \mathbf{P}^f\left(\mathscr{M}(\mathsf{D}^f, \mathbf{T}^f), {\mathsf{I}_\mathsf{p}}^f\right) \tag{4}$$

where $\mathscr{M}(\cdot)$ is a texture mapping function, f is the frame and I_p is an inpainted image estimated from the input image.

3 Results

We tested our method on two examples, a Yacht sailing and an Air balloon gliding through the air. The 3D models were obtained from Internet databases (freely available). After a manual initial alignment, the pose is estimated over frames using edges where both videos present a relatively uniform background. The physical models are built with 192 tetrahedral elements and stiffness parameters of ($E = 450$ Kpa, $\nu = 0.45$)for the Yacht and 288 tetrahedral elements and stiffness parameters of ($E = 25$ Kpa, $\nu = 0.45$) for the Airballoon. These parameters can be tuned as desired by the user, and pre-defined parameters are automatically set according to the 3D model size. We used the framework Sofa [1] for physical simulation.

From the estimated pose, a rigid model is projected on the image to sequentially compensate the background. Indeed inpainting the background to remove the actual object is necessary and needs to be done at each frame.

[1]www.sofa-framework.org

Several manipulations are enabled: stretching, pulling, torsions, rigid transform. The process runs in real-time at 25 fps, making user manipulations interactive.

Fixed constraints are set manually on the physical model and represent the "boat pivot" for the Yacht and the "basket" for the Airballoon example.

Figure 2 shows the obtained results with the Yacht and Airballoon examples.

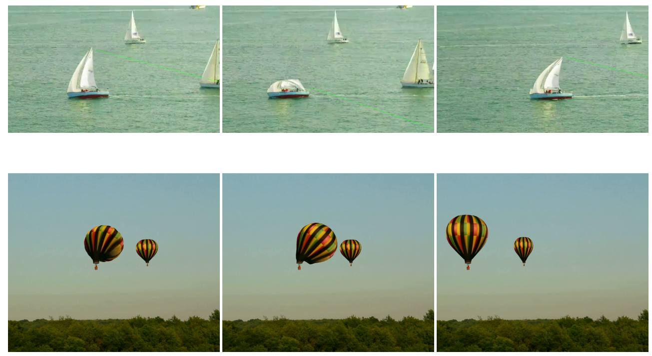

Figure 2: Results obtained on the yacht example and the Air-Balloon example: user-manipulations are possible in a non-rigid manner, where users can pull the objects, deform it in different directions and also change its shape.

4 Conclusion

We presented "Deformed Reality" a new paradigm for augmented reality that enables the user to deform objects in scenes without producing visual breaks usually emanating from 3D mesh overlay in classical AR systems. Using an efficient method to estimate object pose and relying on a physics-based model built on-the-fly, the final composition is rendered sequentially to produce a sequence where objects are deformed and registered in real-time giving sens of manipulating the surrounding of the objects. Several limitations are to be addressed to bring our concept to a well-established technique mainly: (i) real-time continuous inpainting, (ii) robust pose estimation of partially occluded objects and (iii) appearance and illumination estimation from images. With this in mind, we strongly believe that "Deformed Reality" can open new ways in making AR more interactive and enhance user experience with its surrounding.

References

[1] W. H. Chun and T. Höllerer. Real-time hand interaction for augmented reality on mobile phones. In *Intelligent User Interfaces*, pp. 307–314. ACM, 2013. doi: 10.1145/2449396.2449435

[2] N. Kholgade, T. Simon, A. Efros, and Y. Sheikh. 3d object manipulation in a single photograph using stock 3d models. *ACM Trans. Graph.*, 33(4):127:1–127:12, July 2014. doi: 10.1145/2601097.2601209

[3] M. Müller, J. Dorsey, L. McMillan, R. Jagnow, and B. Cutler. Stable real-time deformations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '02, pp. 49–54. ACM, New York, NY, USA, 2002. doi: 10.1145/545261.545269

[4] A. Petit, E. Marchand, and K. Kanani. Augmenting markerless complex 3d objects by combining geometrical and color edge information. In *Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on*, pp. 287–288. IEEE, 2013.

[5] A. Petit, E. Marchand, and K. Kanani. Combining complementary edge, keypoint and color features in model-based tracking for highly dynamic scenes. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 4115–4120. IEEE, 2014.